A Packet Speech Measurement Facility: Final Technical Report

Technical Report
CCA-79-15
January 30, 1979

David A. Low

Computer Corporation of America
575 Technology Square
Cambridge, Massachusetts 02139

Final Technical Report

ARPA Order Number
3331/2/3

Contract Number
MDA903-77-C-0156

Name of Contractor
Computer Corporation of America

Principal Investigator
J. Rothnie, 617-491-3670

Effective Date of Contract
March 11, 1977

Short Title of Work
Packet Speech Measurement Facility

Contract Expiration Date
December 31, 1978

Contract Period Covered by Report
January 1, 1978 - December 31, 1978

This research was sponsored by the
Defense Advanced Research Projects
Agency under ARPA Order No.: 3331/2/3
Contract No.: MDA903-77-C-0156

AD61346

12

(15) MDA903-77-C-0156,
ARPA Order-3331

Computer Corporation of America
575 Technology Square
Cambridge, Massachusetts 02139

(14) CCA-79-15

(6) A Packet Speech Measurement Facility.
(9) Final Technical Report, 1978,
for period ending

(10) David Low
J. Rothrie

(11) 30 Jun 79

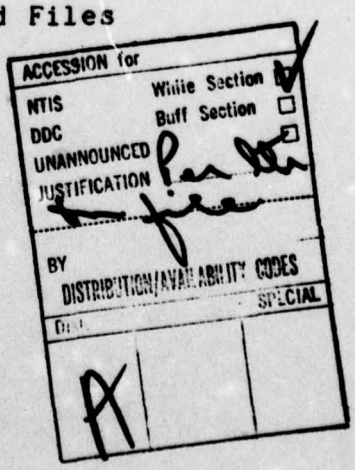
(12) 45 p.

This research was supported by the Defense Advanced Research Projects Agency of the Department of Defense under Contract No. MDA-903-77-C-0156. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

387285

1/B

Table of Contents



ABSTRACT

1. Introduction

The Packet Speech Measurement Facility (PSMF) is an investigative tool designed to be used by researchers in packet network studies. The PSMF facilitates experiments dealing with the timing and composition of packet flow, and will help elucidate the interactions between the conceptual structures of protocol design and the physical exigencies of network implementation.

This report summarizes efforts undertaken by the Computer Corporation of America during the second year of PSMF development: January 1, 1978 to December 31, 1978. These efforts have been aimed at the generalization of existing PSMF functions to serve a larger community of packet network experimenters, and have culminated in the design specification for the augmented features, the implementation of additional protocols, and the installation of some server facilities.

Chapter 2 reviews the design philosophy and structure of the PSMF as it was operational at the end of the first year of development. Chapter 3 reviews the goals and approaches of the second year of development. Chapter 4

describes the procedures for PSMF access, while Chapter 5 details implementation specifics, and reviews the PSMF project. Chapter 6 provides a summary of the disposition of source code and experimental files created during the course of PSMF development.

2. PSMF - First Year Design Review

The operational goal of the first year of PSMF development was to provide a facility whereby Network Secure Communications (NSC) participants could rigorously examine network effects on real time encoded packet speech streams. Three primary functions were designed to satisfy this goal:

1. The provision for recording speech segments along with time stamps reflecting network perturbations.

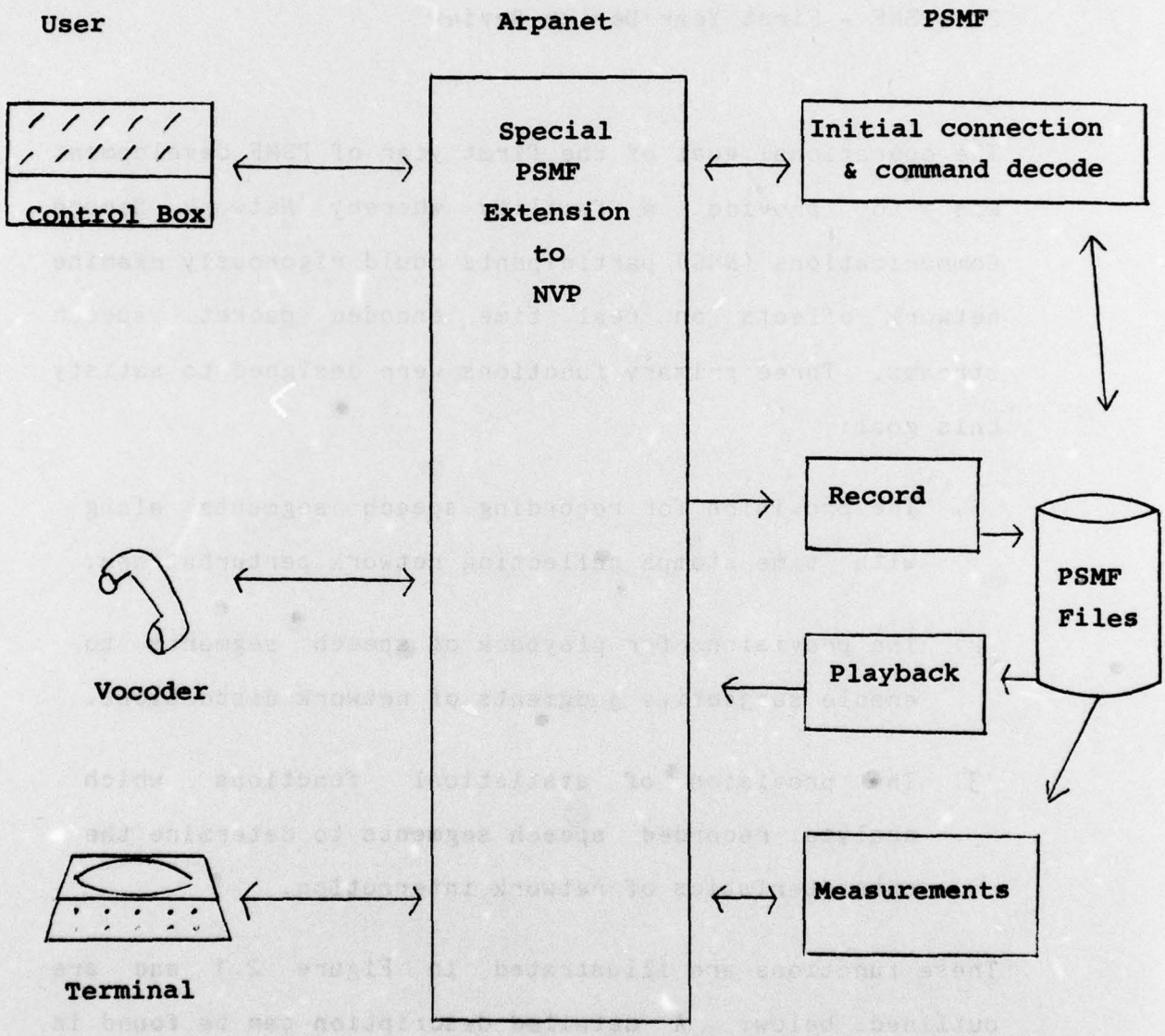
2. The provisions for playback of speech segments to enable subjective judgments of network distortions.

3. The provision of statistical functions which analyze recorded speech segments to determine the characteristics of network interaction.

These functions are illustrated in Figure 2.1 and are outlined below. A detailed description can be found in [LOW].

PSMF - First Year Functions

Figure 2.1



2.1 Recording Function

The PSMF provides a facility for recording a stream of encoded speech messages. Provision is also made to record special PSMF control messages in addition to, or instead of, the speech stream.

The PSMF time-stamps the incoming message stream, classifies it, and later sorts it by user's time stamp. This information is available via the measurement functions described in Section 2.3.

The recording protocol provides a mechanism for:

a. Specifying the file name and access password of the file to be recorded.

b. Conducting voice parameter negotiations as per NVP conventions [COHEN].

c. Time-stamping and recording all elements of the control/voice streams.

d. Terminating the recording session and closing the file.

2.2 Playback Function

A previously recorded file can be opened for playback. The user can selectively specify which elements of the recorded file are to be retrieved.

The playback protocol provides a mechanism for:

a. Specifying the file name and access password of the file to be opened.

b. Specifying the types of messages to be retrieved.

c. Conducting voice parameter negotiations if the user specified the retrieval of voice messages.

d. Transmitting stored packets at a rate determined by negotiated parameters.

e. Permitting file closure or rescan at the user's option.

2.3 Measurement Functions

A previously recorded file can be opened for measurement. The user can then specify the file which will receive the measurement data and the measurement function. On completion of the measurement, the resulting file can be played back.

The file created during execution of a measurement function may itself be subjected to further measurement. Measurements may thus be compounded. For instance, a file of relative delay times might be created, followed by another measurement creating a file of its histogram.

The measurement protocol provides mechanism for:

a. Specifying the input file name and access password.

b. Specifying the output file name and access password.

c. Specifying the measurement function and associated parameters.

The available functions are illustrated in Figure 2.2 and are outlined below.

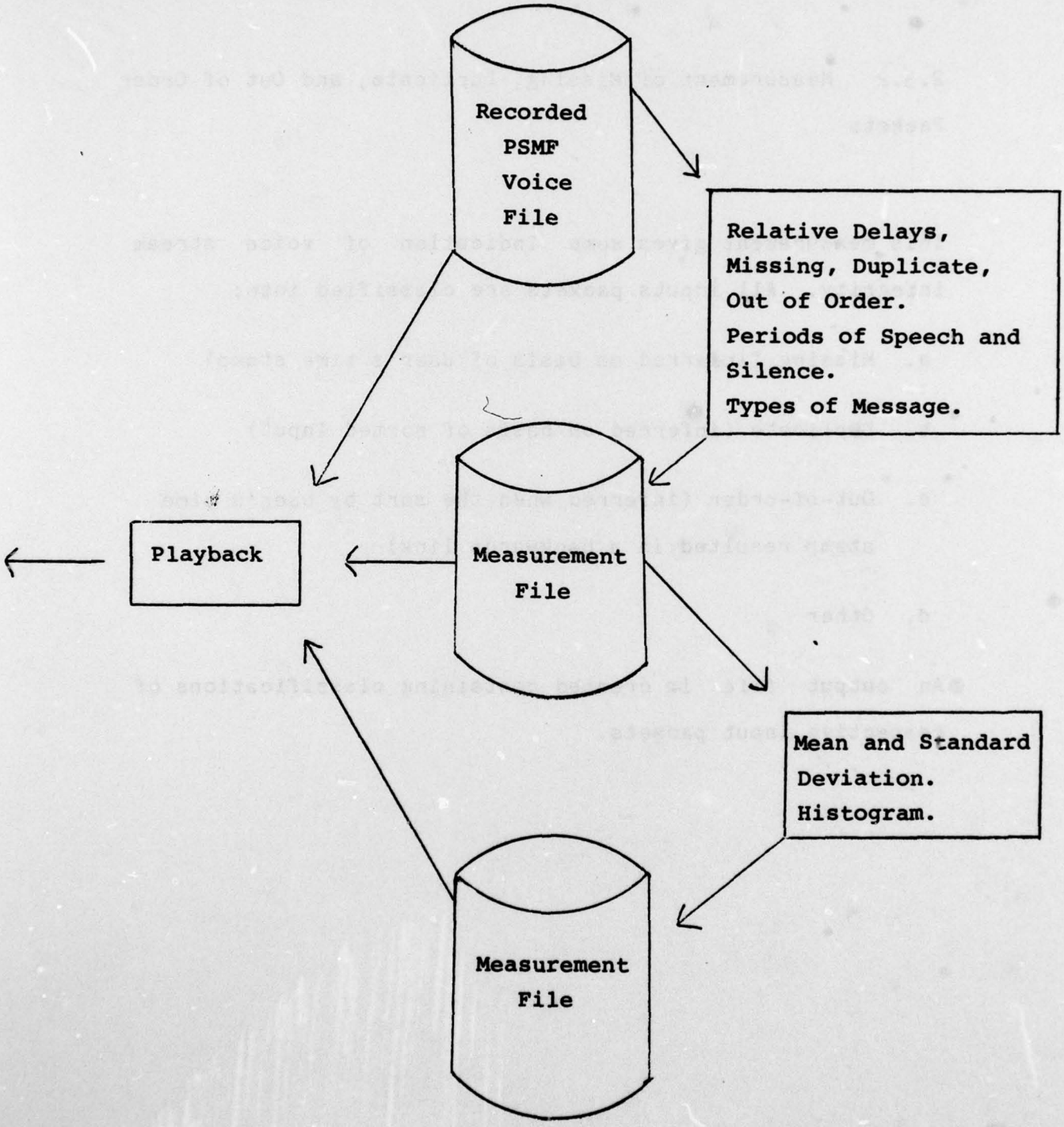
2.3.1 Measurement of Relative Delays

The absence of an absolute time standard means that delay times have to be measured relative to some standard. The PSMF relative delay function produces a file of delays relative to that experienced by the first voice packet:

relative delay = (PSMF timestamp for this message -
PSMF timestamp for first message) -
(user's timestamp for this message -
user's timestamp for first message)

An output record containing this value is created for each input voice packet.

PSMF - First Year Measurement Functions
Figure 2.2



2.3.2 Measurement of Missing, Duplicate, and Out of Order Packets

This measurement gives some indication of voice stream integrity. All inputs packets are classified into:

a. Missing (inferred on basis of user's time stamp)

b. Duplicate (inferred on basis of sorted input)

c. Out-of-order (inferred when the sort by user's time stamp resulted in a backwards link)

d. Other

An output file is created containing classifications of respective input packets.

2.3.3 Measurement of Periods of Speech and Silence

This measurement function scans the input file for indications that the user stopped vocoding during periods of silence. The length of these periods, as well as the previous and subsequent voiced periods, is computed and recorded in the output file.

2.3.4 Computation of Mean and Standard Deviation

This is one of two measurement functions which normally use a previously created measurement file as input. The input file is scanned for records of a particular type (that type created by measurement functions). A field in each such record is extracted and subjected to a conclusion of the input file, an output file is created consisting of three records specifying mean, standard deviation, and count.

2.3.5 Computation of Histogram

This measurement function also normally uses a previously created measurement file as input. When the user specifies this particular measurement, he also states the lower bound and interval size of the histogram. The measurement then scans the input file for data records of a special computational type. A histogram of data contained in these records is computed and written to the output file on conclusion of the input.

3. PSMF - Second Year Design Review

It should be noted again that the original PSMF design, as summarized above, restricted analyses to packet streams transmitted under Network Voice Protocol (NVP) [COHEN]. In addition, an extension of this protocol was necessary to access the PSMF at all. These restrictions limited the scope of PSMF utilization.

The primary goal of the second year of PSMF development was to provide a generalized facility, free of the restrictions described above. The steps taken to achieve this goal were as follows:

1. Generalization of the PSMF recording facility to transcribe packet streams other than those sent under the NVP.

2. Provision of facilities whereby users could schedule experiments, subject them to measurements, and recover experimental files using common network protocols.

3. Incorporation of a Transmission Control Program. Initial work was performed on this task, which was intended to open the PSMF to the internetworking environment.

These approaches are described in detail below.

3.1 Generalization of the Recording Facility

The original PSMF recording facility was designed to transcribe packet streams sent under the Network Voice Protocol. There are other protocols, both developmental and standard, for which an experimental recording facility would be useful. A generalization of the PSMF recording function was undertaken in order to provide this facility.

There are two ways in which the PSMF can participate in a recording session. One situation involves the PSMF as an active participant in a dialog or conference, e.g. an NVP of NVCP session. In this case the PSMF implements the protocol of interest, and experiments can be performed to evaluate the effects of the user's parameters on packet flow.

In the other case the PSMF may act as a non-participatory intermediary for a packet stream. In this mode the PSMF functions as a recording and forwarding station, and needs to know only those elements of the protocol which enable it to perform this function unobtrusively.

The generalization of the PSMF recording facility includes provision for both these situations. The PSMF can actively participate in communication streams involving the NVP and NVCP. For recording sessions under an arbitrary protocol, the Record and Relay (RAR) facility gives experimenters PSMF access without the need for time-consuming and expensive software specialization.

Each of these enhancements to the PSMF is discussed below.

3.1.1 Network Voice Conferencing Protocol

The PSMF has been augmented to accommodate the Network Voice Conferencing Protocol (NVCP) [COHEN]. An extension to the NCVP was provided to allow a file name/password specification in the "please join my conference" message. On receiving this message, the PSMF:

a. Opens the requested file

b. Initiates a connection with the conference chairman

c. Carries on voice negotiations with the chairman

d. Records all such negotiations and ensuing voice messages in a format amenable to the playback and measurement functions of the standard NVP accessible PSMF.

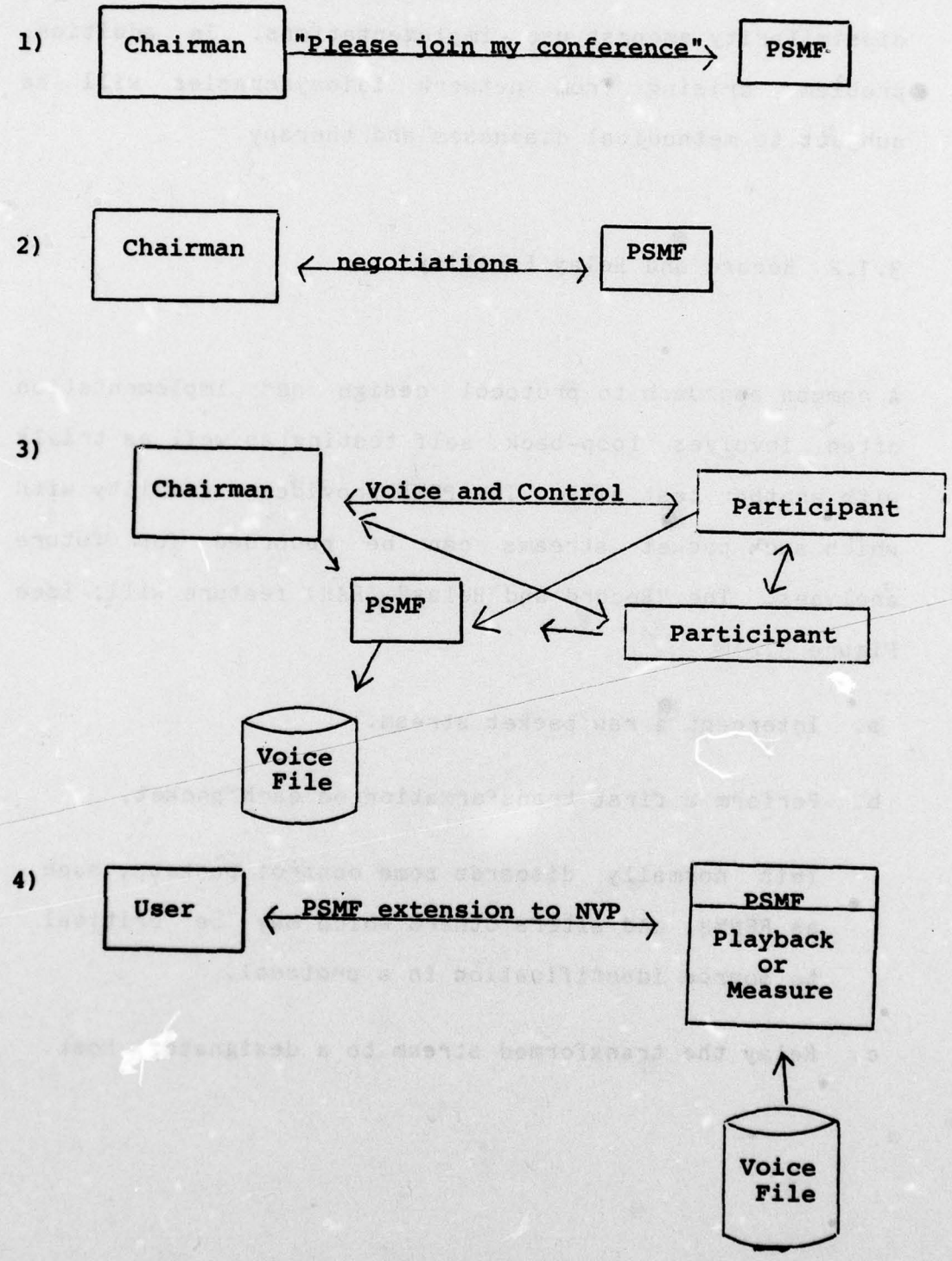
e. Closes the file on receipt of a "good-bye" message.

This procedure is illustrated in Figure 3.1.

The incorporation of an NVCP facility in the PSMF provides experimenters with a variety of services. Aside from the obvious use of the facility as a "stenographer", the NCVP

NVCP Operations

Figure 3.1



recorder will help resolve problems resulting from a dissimilarity amongst user implementations. In addition, problems arising from network idiosyncrasies will be subject to methodical diagnoses and therapy.

3.1.2 Record and Relay Facility

A common approach to protocol design and implementation often involves loop-back self testing as well as trials with another test site. The PSMF provides a facility with which such packet streams can be recorded for future analyses. The "Record and Relay" (RAR) feature will: (see Figure 3.2)

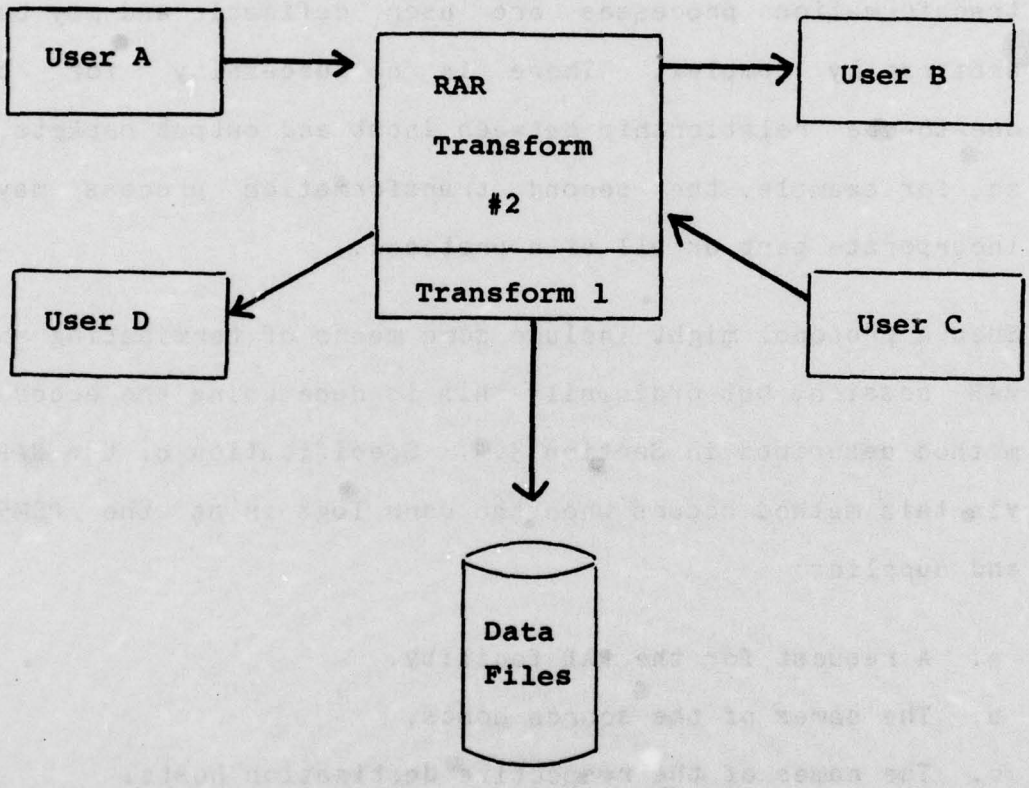
a. Intercept a raw packet stream.

b. Perform a first transformation on each packet.

   This normally discards some control packets, such as RFNMs, and alters others which may be critical to source identification in a protocol.

c. Relay the transformed stream to a designated host.

Record and Relay Facility - Packet Flow
Figure 3.2



Users A, B, C, D need not be distinct.

(In particular, A=D, B=C or A=B=C=D is allowed.)

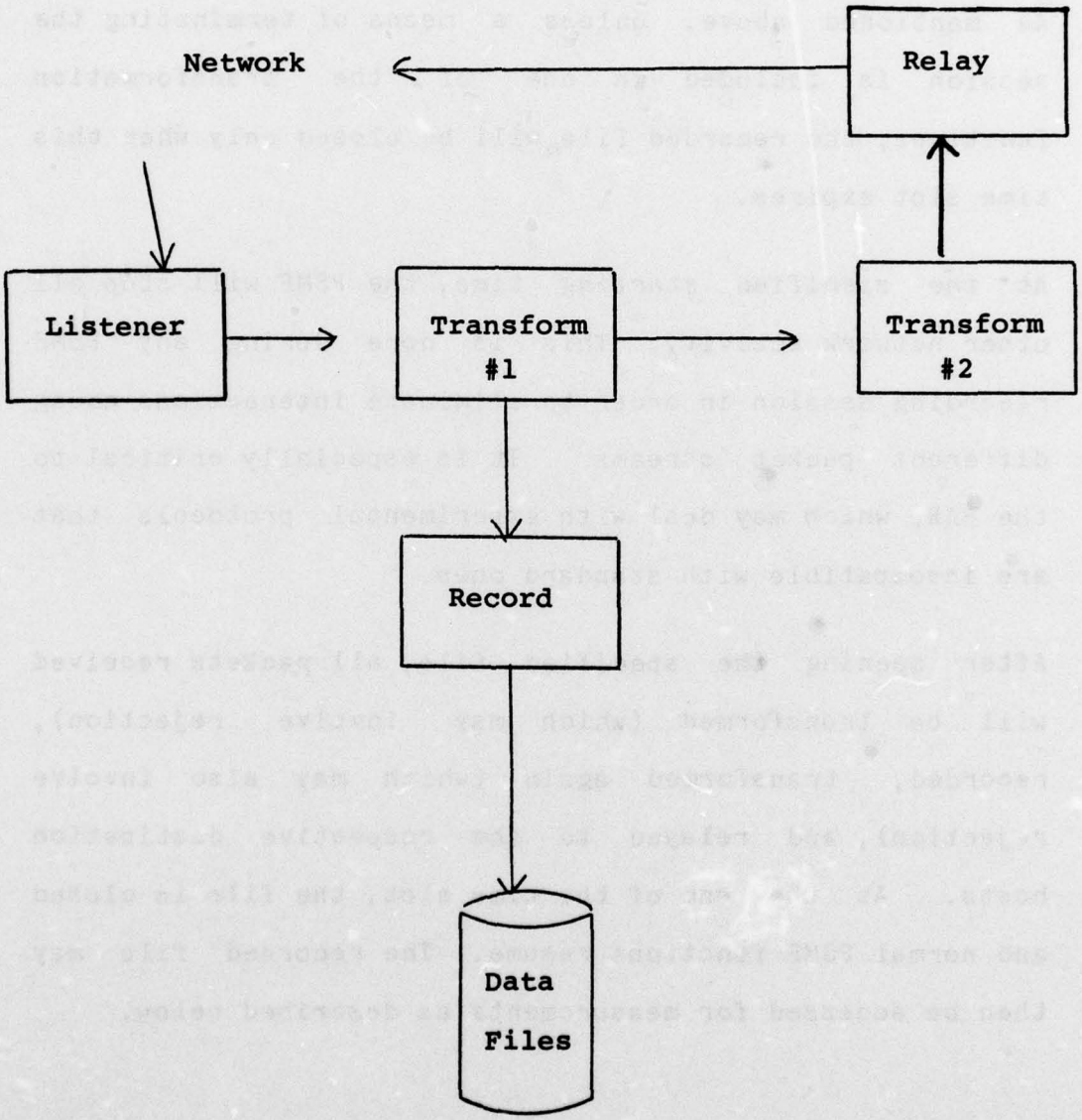
The RAR is implemented as a series of four processes (see Figure 3.3), each of which completes its own manipulation of a packet before passing its address onto the next process. In particular, it should be noted that the two transformation processes are user definable and may be arbitrarily complex. There is no necessity for a one-to-one relationship between input and output packets, so, for example, the second transformation process may incorporate part or all of a protocol.

Such a protocol might include some means of terminating an RAR session, but ordinarily this is done using the access method described in Section 3.4. Specification of the RAR via this method occurs when the user logs in at the PSMF and supplies:

a. A request for the RAR facility.
b. The names of the source hosts.
c. The names of the respective destination hosts.
d. The name of the file to be recorded.
e. The names of the transformation functions.

These are standard library functions or are created by the user during normal PSMF operation. They are constructed using the same facilities as measurement functions.

Record and Relay Facility - Process Communication
Figure 3.3



f. The time slot during which the recording will take place.

As mentioned above, unless a means of terminating the session is included in one of the transformation functions, the recorded file will be closed only when this time slot expires.

At the specified starting time, the PSMF will stop all other network activity. This is done during any PSMF recording session in order to eliminate interactions among different packet streams. It is especially critical to the RAR, which may deal with experimental protocols that are incompatible with standard ones.

After opening the specified file, all packets received will be transformed (which may involve rejection), recorded, transformed again (which may also involve rejection), and relayed to the respective destination hosts. At the end of the time slot, the file is closed and normal PSMF functions resume. The recorded file may then be accessed for measurements as described below.

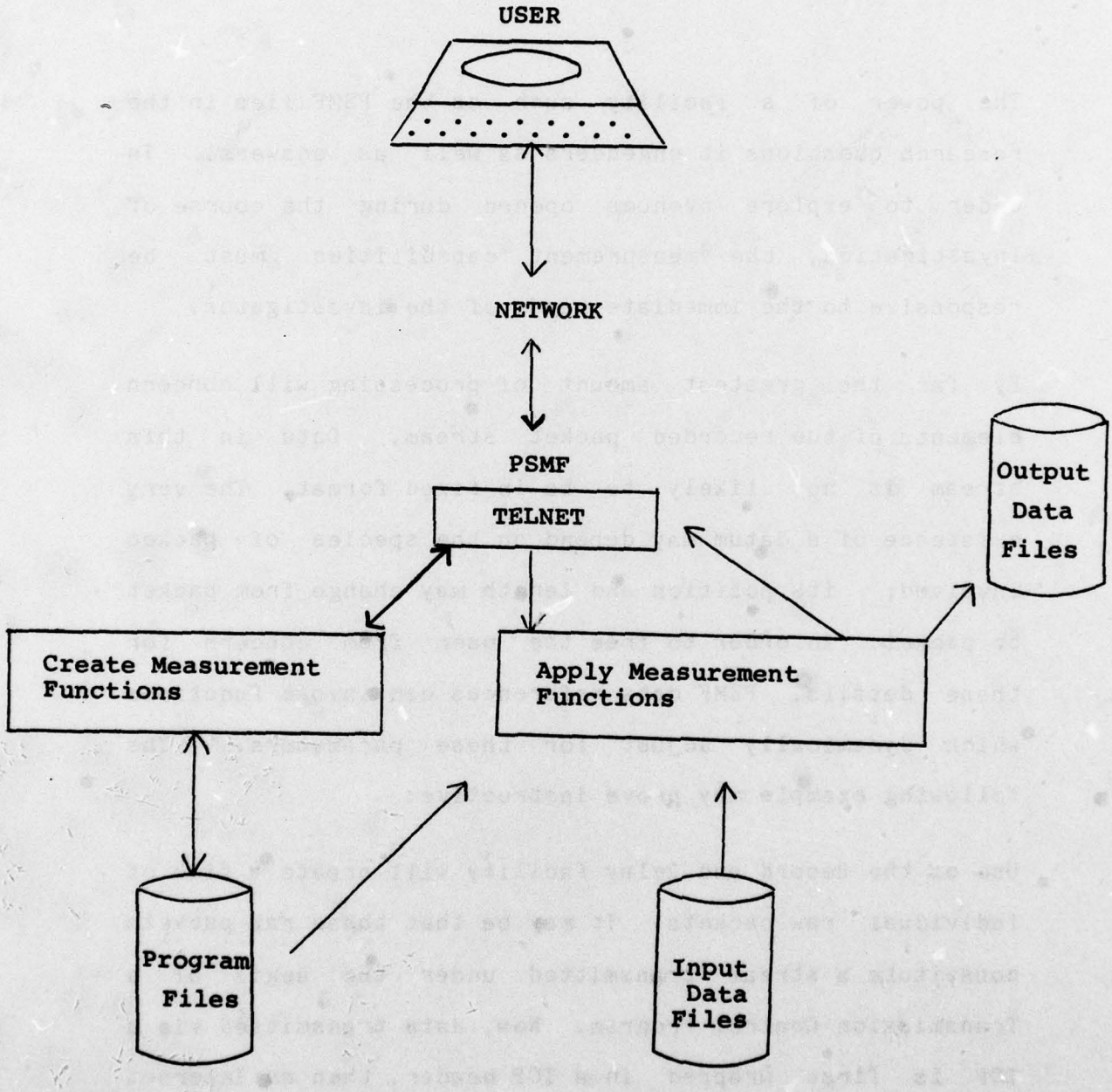
3.2 Measurement System

The power of a facility such as the PSMF lies in the research questions it engenders as well as answers. In order to explore avenues opened during the course of investigation, the measurement capabilities must be responsive to the immediate needs of the investigator.

By far the greatest amount of processing will concern elements of the recorded packet stream. Data in this stream is not likely to be in fixed format. The very existence of a datum may depend on the species of packet involved; its position and length may change from packet to packet. In order to free the user from concern for these details, PSMF data references can invoke functions which dynamically adjust for these parameters. The following example may prove instructive:

Use of the Record and Relay facility will create a file of individual raw packets. It may be that these raw packets constitute a stream transmitted under the aegis of a Transmission Control Program. Now, data transmitted via a TCP is first wrapped in a TCP header, then an Internet header, and finally (for the Arpanet) in an Arpanet

User Defined Measurement Facility Figure 3.4



host-to-host header. (The subnet header can be considered to be invisible for our purposes.) The Internet header and TCP header are both of variable length, and in addition, the TCP packet may be fragmented.

PSMF measurement functions can be constructed to reflect this hierarchy:

GETRAW (BUFFER_AREA,BUFFER_LENGTH,PACKET_SIZE)

may try to get the next raw packet in a file. If the file is not at "end of file", and if BUFFER_LENGTH is large enough for it, GETRAW can return a "true" indication, transfer the data into BUFFER_AREA, and return the actual PACKET_SIZE.

The function

GETINTERNET (BUFFER_AREA,BUFFER_LENGTH,PACKET_SIZE)

may try to get the next Internet packet in a manner analogous to GETRAW above. In fact, GETINTERNET could be constructed from GETRAW calls.

In a similar way,

GETTCP (BUFFER_AREA,BUFFER_LENGTH,PACKET_SIZE)

can be constructed from GETINTERNET calls to assemble a TCP packet.

In addition, the PSMF library can be formulated to contain function definitions for all elements of standard headers, so that, for instance, the function

```
URGENT (BUFFER_AREA)
```

can return the value of that bit in the TCP header.

So, if the user wants to count all occurrences of the "urgent" bit, the following function would suffice:

```
SUM = 0;
WHILE (GETTCP(BUFFER_AREA,BUFFER_LENGTH,PACKET_SIZE))
      SUM = SUM + URGENT (BUFFER_AREA);
```

When the definitions of the Internet and TCP headers are changed, only the low-level library definitions need be altered: a task which may be performed by PSMF personnel. The users' functions remain the same.

There may, of course, be a variety of protocols under consideration, some of which might be slight modifications of others. The PSMF provides the user with the ability to create and specify sub-libraries of protocol dependent definitions.

The discussion of the Record and Relay Facility in Section 3.1.2 mentioned that the "transformation" functions were created in the same way as measurement functions. It can

be noted here that this equivalence extends to use of the same library functions. Two examples follow:

The second RAR transform usually works on a packet by packet basis. Some packets, such as RFNMs, are discarded as being unsuitable for relaying. Others may require a modification to headers before forwarding. Internet packets, for instance, require some such computation:

a. The Arpanet header has to be transformed to reflect the new destination host/gateway.

b. The Internet header has to be transformed to indicate the new source and destination networks and addresses.

c. The Internet header checksum has to be recomputed.

All these tasks necessitate access to fields in headers which are subject to reformatting. The transformation function can thus employ the same low-level library definitions to access these headers as the measurement functions do, thus obviating the need for high-level reprogramming for low-level changes.

In another example, although the second RAR transform usually works on a packet-to-packet basis, it is not

constrained to do so. Thus, as in the measurement function outlined above, the transform function could collect an entire TCP packet before forwarding. Given some element of retransmission protocol, it is not unreasonable to think of the RAR, in this case, as a gateway.

3.3 Transmission Control Protocol

It is probable that the networking environment of the future will be dominated by the emergence of issues engendered with internetwork communications. Such communications, involving as they do a disparate and de-centralized collection of resources and responsibilities, offer hazards which cannot be anticipated simply from Arpanet experience. The PSMF offers an ideal site from which to probe networks and hosts to help resolve problems concerning packet timing and integrity.

As an initial step in this direction, CCA started work on implementation of a Transmission Control Protocol (TCP) for the PSMF machine. The intended uses for this implementation were:

a. As a basis for higher level protocols (e.g. TELNET) with which future users could access PSMF facilities.

b. As a full-time test facility for use by other TCP implementors.

c. As a measurement arm of the PSMF, to permit analyses of packet streams subjected to this protocol.

4. PSMF Access Procedures

There are two important modes of PSMF access. The first entails a recording session. Because packet timing is often important in such a session, and because of the difficulty of working simultaneously with different host-to-host protocols, all other PSMF access is locked out during recording. The procedure for this access is as follows:

a. The user logs in on the PSMF using "new" TELNET (currently on top of NCP). User names and passwords are arranged with CCA personnel.

b. The user requests a reservation, specifying the time slot and the nature of the reservation (protocol to be used, file name to be created, etc.).

c. If the PSMF detects no conflict, the reservation is made.

d. Before the time slot reserved, the PSMF requests all other users to log off. At the beginning of the slot, all remaining network processes are terminated.

e. The necessary software is installed and the recording session begins.

f. At the end of the time slot, the recording session is terminated.

This eliminates any need for a special protocol extension, but if a user should wish to establish such an extension to enable remote session termination, he may do so.

g. Standard network software is reinstalled and users may log in for measurements.

The second kind of PSMF access involves the creation and use of measurement transformation (for the RAR) functions. The procedure for such access is:

a. The user logs in on the PSMF using TELNET. Again, user names and passwords are arranged with PSMF personnel.

b. If the PSMF is not in a recording session, the user can specify the measurement to be made or construct his own measurement function. Such construction entails specifications of the libraries to be included and the functions and arguments to be employed.

c. Before the time a recording session is scheduled, the user will be asked to log off and will be notified of the session duration. Information about the recording schedule will be available to each user at any time, thus enabling him to make decisions regarding his own needs for processer time.

5. PSMF Operations

5.1 Implementation Specifics

The PSMF as implemented in the first year of development has been operational during the period covered in this report. Scheduling has been by request, and some of the experiments which have been performed are described in the next section.

Second year efforts included:

a. Adaptation of the PSMF to the Network Voice Conferencing Protocol.

b. Conversion of Air Force Systems Control (AFSC) Arpanet software to RSX-11M/V3 operating system.

c. Provision of facilities for user-defined measurements. A "C" compiler for RSX-11M was obtained, and several libraries and analytic routines made available to users.

d. Implementation of the Record and Relay (RAR) facility.

e. Implementation of a remote scheduling system for PSMF recording sessions.

f. Initial work on the implementation of a Transmission Control Program (TCP) running under RSX-11M version 3.1.

5.2 Experiments

While the above second-year efforts were underway, the PSMF as implemented in the first year of development was available for operation by request, and several experiments were performed, in conjunction with ISI, using the PSMF extension to the NVP.

a. Experiments were undertaken to test the network handling of packet streams of varying size and frequency [LOW]. These experiments demonstrated non-linear relationships between such parameters and the number of missing or out-of-order packets.

b. In concert with the Network Control Center, experiments were performed to investigate the relationships between IMP buffering procedures and delays on subtype 3 packets [COLE]. These experiments revealed a sharp and important relationship which has far-reaching implications for packet voice communications.

c. Experiments using the PSMF as a participant in NVCP conferences revealed peculiarities in some users' protocols that helped explain difficulties in communication. In this case, the PSMF was used in the role of a protocol debugger.

6. Summary and Disposition

This report has described the work performed at CCA on the Packet Speech Measurement Facility during the second year of its development.

A review of the initial design considerations was followed by an exposition of the motivations guiding the development of the second phase. A description of the implementation approach followed. This report concludes with an outline of the disposition of the software and data accumulated during the contract period.

The major divisions of software and data files are:

1. Source and data associated with the NVP and NVCP aspects of the PSMF.
2. Source associated with the AFSCNET conversion.
3. Source associated with the Record and Relay (RAR) facility.
4. Source associated with the initial work in implementation of a Transmission Control Protocol (TCP) under RSX-11M version 3.1.

The nature of access to these files depends on the status of the PSMF host (CCA-SPEECH). The following descriptions assume online access via the Arpanet using standard network protocols (FTP, New TELNET) over NCP. Such access requires an account name, which will be given by CCA personnel on request.

In the event that CCA-SPEECH becomes unavailable, or if a user prefers files in a magtape format, application must be made to the Sponsored Research Division of CCA.

6.1 NVP and NVCP Associated Files

Table 6.1 lists the important source files for NVP and NVCP PSMF processes. In addition, there are a host of support files to assemble and build the various modules. These can be listed using the RSX PIP facility. These files are all under the UICs [350,10] and [351,10].

There are about 220 online PSMF data files containing about 10 million bytes of data. These are all under UIC [350,77], can be listed using PIP and transferred using CCA-SPEECH's server FTP.

In addition, there are about 90 offline PSMF data files containing about 5 million bytes. These files contain

data for the daily experiments described in [LOW] and can be obtained (or put on line) on application to CCA.

The format for these data can be found in [LOW].

6.2 AFSCNET Conversion Files

A major part of the second PSMF year effort was apportioned to a conversion of AFSCNET NCP, server TELNET and server/user FTP from RSX-11 on version 2 to version 3.1 The responsibility for this software (now called SAMNET) still lies with the Air Force Systems Command, and application must be made to them before CCA can release any of the modified software. At the time of this writing, applications should be submitted to Captain Arthur Lundquist (Lundquist @ BBN-TENEX).

PSMF Processes Table 6.1

Pathname	Comments
[350,10] NVP375.MAC	test process
[350,10] NVP357.MAC	PSMF root
[350,10] COMPSM.MAC	PSMF control process
[350,10] CMNPSM.MAC	data definition and shared data
[350,10] LBRPSM.MAC	library (shared) functions
[350,10] PARSER.MAC	RSX parser for PSMF commands
[350,10] SRT401.MAC	asynchronous sort process
[350,10] MEASUR.MAC	PSMF measurement overlay
[350,10] PLAYBA.MAC	PSMF playback overlay
[350,10] NVP377.MAC	PSMF listener process
[350,10] RECORD.MAC	PSMF record overlay
[350,10] FILES.MAC	PSMF file interface
[351,10] VCP357.MAC	NVCP PSMF record root
[351,10] ECH357.MAC	PSMF echo process

6.3 Record and Relay Associated Files

The CCA Record and Relay (RAR) facility software was written largely in the "C" language (the compiler for RSX was supplied by Yourdon, Inc.). Included as part of this effort were processes designed to give a remote user facilities with which to schedule exclusive access to PSMF recording capabilities. Table 6.2 lists the major source files associated with this software. A description of RAR and its processes may be found in [310,10] RARUSER.TXT.

Again, there are many ancillary files concerned with assembling and linking the various processes. These may be accessed using the PIP process of RSX.

6.4 TCP Associated Files

The CCA implementation of a Transmission Control Protocol (TCP) was not completed during the period of this contract. The effort may be ongoing, but the extant software is not operational or documented. Application for this software should be made to the CCA Sponsored Research Division.

RAR and Scheduling Processes Table 6.2

Pathname	Comments
[354,10] SCHRAR.CCC	scheduling process
[354,10] CONRAR.CCC	RAR control process
[354,10] RECRAR.CCC	RAR record process
[354,10] TR1.CCC	RAR first transform process
[354,10] TR2.CCC	RAR second transform process
[354,10] SCHDMN.CCC	Scheduling daemon process
[354,10] EXSCHE.CCC	Schedule examiner process
[354,10] ENDRAR.CCC	RAR terminating process
[354,10] RFILES.MAC	RAR files interface
[354,10] CEXEX.MAC	"C" to RSX executive interface
[354,10] CLBR.CCC	RAR library
[354,10] PKTSCM.CCC	RAR statistics process

References

[LOW]
Low, D. "A Packet Speech Measurement Facility: Final Technical Report, 1977" Technical Report No. CCA-78-05, Computer Corporation of America, 575 Technology Square, Cambridge, Massachusetts 02139, February 1978.

[COHEN]
Cohen, D. "Specifications for the Network Voice Protocol" ISI/RR-75-39, Information Sciences Institute, University of Southern California, March 1975.

[COHEN 2]
Cohen, D. "The Network Voice Conferencing Protocol (NVCP)" Information Sciences Institute, University of Southern California, February 1976.

[COLE]
Cole, R. Report to the Network Secure Communications Conference, May 1978.